REMARKS

The above amendments to the above-captioned application along with the following remarks are being submitted as a full and complete response to the Official Action dated June 3, 2004. In view of the above amendments and the following remarks, the Examiner is respectfully requested to give due reconsideration to this application, to indicate the allowability of the claims, and to pass this case to issue.

Status of the Claims

Claims 1-8 are under consideration in this application. Claims 1-8 are being amended, as set forth above and in the attached marked-up presentation of the claim amendments, in order to more particularly define and distinctly claim Applicants' invention.

Additional Amendments

The claims are being amended or added to correct formal errors and/or to better disclose or describe the features of the present invention as claimed. Further, the specification is being amended to correct various informal errors and to clarify the disclsoure of the invention as claimed. Applicants hereby submit that no new matter is being introduced into the application through the submission of this response.

Formality Rejection

The Title of the Invention was objected to as being non-descriptive, and the Examiner has requested a new title which better describes the invention. Claims 1-8 were rejected under 35 U.S.C. § 112, second paragraph, as being indefinite. As indicated, the Title and the claims have been amended as required by the Examiner. Accordingly, the withdrawal of the outstanding informality rejection is in order, and is therefore respectfully solicited.

Prior Art Rejection

Claims 1-4 and 6-8 were rejected under 35 U.S.C. § 102(b) on the grounds of being anticipated by a publication of Intel Corp.: IA-64 application Developer's Architecture Guide, May 1999 (hereinafter "Intel"). Claim 5 was rejected under 35 U.S.C. 103(a) on the grounds of

being unpatentable over Intel. These rejections have been carefully considered, but are most respectfully traversed.

The processor of the invention (Fig. 1; p. 9), as now recited in claim 1, comprises: a register file 213 (Fig. 3) including a plurality of registers R0-R63 assigned with register numbers, each of the registers storing operand data; a plurality of operation pipes, each operation pipe for executing in parallel one kind of operation (e.g., "*data alignment (ALN) 211, multiply (MUL) 212, integer operation (INT), load/store (LD/ST) and branch (BRA)*" p. 9, 2nd paragraph) on a plurality of the operand data; a decoder 202 (Fig. 4) for decoding a register designating field ***m*** of an instruction code 201, the register designating field having a register number stored therewith, the decoder further for generating signals designating register numbers based on the register number of the register designating field, the designated register numbers being consecutive with the register number of the register designating field (e.g., Source 1 and Source 2 in Fig. 23; "*4 consecutive registers can be designated at one time*" p. 9, line 21); and a control circuit for sending operand data stored in the registers corresponding to the designated register numbers to at least one of the operation pipes such that the at least one of the operation pipes executes in parallel the one kind of operation associated therewith on the operand data sent from the corresponding designated registers pipes (e.g., multiply (MUL) 212) for executing the operation therein in parallel (e.g., A*E, B*F, C*G, D*H).

The present invention as recited in claim 2 is directed to a processor comprising: a register file including a plurality of registers assigned with register numbers; a plurality of operation pipes, each operation pipe for executing in parallel one kind of operation on a plurality of the operand data so as to generate operation result data; a decoder for decoding a register designating field of an instruction code, the register designating field having a register number stored therewith, the decoder further for generating signals designating register numbers based on and consecutive with the register number of the register designating field, and the registers corresponding to the designated register numbers being designated for storing the operation result data; and a control circuit for sending the operation result data from at least one of the operation pipes to the corresponding designated registers.

As recited in claim 3, the present invention is directed to a processor comprising: a register file including a plurality of registers assigned with register numbers, each of the registers for storing at least one of operand data and operation result data; a plurality of operation pipes, each operation pipe for executing in parallel one kind of operation on a plurality of the operand

data so as to generate the operation result data; a first decoder for decoding a first register designating field of an instruction code, the first register designating field having a first register number stored therewith, the first decoder further for generating signals designating source register numbers based on and consecutive with the first register number; a second decoder for decoding a second register designating field of the instruction code, the second register designating field having a second register number stored therewith, the second decoder further for generating signals designating result register numbers based on and consecutive with the second register number; and a control circuit for sending the operand data stored in source registers corresponding to the designated source register numbers to at least one of the operation pipes such that the at least one of the operation pipes executes in parallel the one kind of operation associated therewith on the operand data and for sending the operation result data obtained from the at least one operation pipe to result registers corresponding to the designated result register numbers.

Under a conventional permutation instruction, such as those shown on page 7-130 of Intel, when permutation operation is carried out with regard to, for example, two pieces of SIMD data of 16 bit x 4 pieces, two pieces of 64-bit wide registers for storing result are needed; however, only one register can be designated and accordingly, separate instructions are prepared for an upper bit portion and a lower bit portion of the permutation result. The invention, however, designates a plurality of registers via the register designating field, e.g., four source registers as shown in Fig. 16, permutation operation of the upper bit portion and the lower bit portion is carried out at one time and two set portions can simultaneously be operated (p. 15, last paragraph to p. 16, 1st paragraph).

Applicants respectfully contend that Intel fails to teach or suggest such register designating fields ("RField") contained in an instruction code to be decoded by a decoder which make it possible to read operand data from or to write operation result data into a plurality of designated registers ("Rs"), i.e., "1 RField: Rs" to greatly enhance the data processing capability according to the invention. Further, the register designating field of the invention supports a plural-to-plural permutation scheme for reading operand data from plural source registers ("SRs") into plural result registers ("RRs") after the operation as shown in Figs. 16, 20 & 23, i.e., "SRs: RRs."

The Examiner interpreted each of the ***write ports*** of the source register GR r2 in Fig. 7-23 on page 7-117 in Intel as a "register designating field" (p. 4, lines 11-13 of the outstanding office

action). However, Intel's decoder (page 6-2) only decodes IA-64 instructions, rather than any "register number embedded in a register designating field ***m*** contained in an instruction code 201" according to the invention. Further, Intel's decoder does not thereby designate consecutive registers or at least their corresponding register numbers consecutively for an operation.

In particular, Intel's processor architecture handles data stored in only one register by the use of one register designating field. Intel's instruction code permits one to designate 3 registers r1, r2 and r3 through up to 3 register designating fields. The example shown in Intel's pages 7-117, 7-130 indicates that one register designating field handles 8-bit elements within one register GR r2. The example shown in Intel's page 7-131 indicates that one register designating field handles 16-bit elements within one register GR r2. Intel only reads data from "***a single source register***" GR r2 (p. 7-130, line 3), i.e., "1 RField; ***1 R***," rather than i.e., "1 RField: ***Rs***" as according to the invention. Contrary to the Examiner's assertion that "*the single source register GR r2 designates eight consecutive registers* (p. 5, paragraph No. 8), " Intel's register designating field merely designates 8-bit elements in the single source register GR r2.

The actual register designating field in Intel only supports an one-to-one permutation scheme for reading data from a single source register GR r2 (p. 7-130, line 3) and writing data into a single result register GR r1, i.e., "**1** SR: **1** RR," rather than ""SRs: RRs" according to the invention. Fig. 7-24 on page 7-130 only shows 5 permutation schemes for 8-bit elements between ONE source register GR r2 and ONE result register GR r1. Fig. 7-25 on page 7-131 only shows 4 permutation schemes for 16-bit elements between ONE source register GR r2 and ONE result register GR r1. Therefore, the processor architecture described in present claims 1-3 is essentially different from that of Intel's.

Accordingly, the present invention as now recited in the independent claims 1-3 is distinguishable and thereby allowable over the rejections raised in the Office Action. The withdrawal of the outstanding prior art rejections is in order, and is respectfully solicited.

Priority Claim

Regarding the Examiner's comments regarding the possible improper certified copy of the Japanese Priority Application No. 2000-340239 under 35 USC §119(b), attached is a copy of the priority application that Applicants had submitted with the filing of of the application on September 4, 2001. The priority application as filed did in fact have a ribbon indicating that it was a certified copy of the original priority application. Therefore, Applicants will submit that

a proper certified copy of Japanese Priority Application No. 2000-340239 under 35 USC §119(b) had been filed and that the scanned image of the document in the possession of the Examiner was not the fault of the Applicants nor of the undersigned representative.

Conclusion

In view of all the above, clear and distinct differences as discussed exist between the present invention as now claimed and the prior art reference upon which the rejections in the Office Action rely, Applicant respectfully contends that the prior art references cannot anticipate the present invention or render the present invention obvious. Rather, the present invention as a whole is distinguishable, and thereby allowable over the prior art.

Favorable reconsideration of this application is respectfully solicited. Should there be any outstanding issues requiring discussion that would further the prosecution and allowance of the above-captioned application, the Examiner is invited to contact the Applicant's undersigned representative at the address and phone number indicated below.

Respectfully submitted,

Stanley P. Fisher
Registration Number 24,344

Juan Carlos A. Marquez
Registration Number 34,072

REED SMITH LLP
3110 Fairview Park Drive, Suite 1400
Falls Church, Virginia 22042
(703) 641-4200

September 16, 2004

SPF/JCM/JT